



哈爾濱工業大學 (深圳)  
HARBIN INSTITUTE OF TECHNOLOGY

# 实验报告

开课学期: 2021 春季  
课程名称: 数据库系统  
实验名称: 微博系统设计与实现  
实验性质: 设计型  
实验学时: 2 地点: T2210  
学生班级: 1801107  
学生学号: 180110710  
学生姓名: 陈妮娜  
评阅教师:  
报告成绩:

实验与创新实践教育中心制

2021 年 1 月

# 1 实验环境

请填写用到的操作系统和主要软件版本。

- 操作系统：win10
- 主要软件：JetBrains PyCharm Community Edition 2018.3.4 x64,MySQL 8.0, Microsoft Edge

# 2 实验过程

## 2.1 系统功能

请结合文字、表格、截图等方式，清晰描述系统的功能。亮点功能请用\*标志。

- 发布微博



当前用户名显示：



编辑微博，选择模块：



发表后可在微博广场上查看。

➤ 评论微博：



amy

今天很开心!

2021-04-24 23:33:31

转发

评论

点赞

点击评论按钮可进入评论页面：

发表评论

请输入内容

发表

发布评论，使用两个不同的账号：



amy

关注

今天很开心!

2021-04-24 23:33:31

转发

评论

点赞 (0)

评论列表

amy

2021-04-24 23:37:39


开心开心

jack

2021-04-24 23:37:57

发生什么好事了?

➤ 点赞微博：



amy

今天很开心!

2021-04-24 23:33:31

转发

评论

点赞 (2)

评论列表

amy

2021-04-24 23:37:39

开心开心

jack

2021-04-24 23:37:57

发生什么好事了?

- Jack 和 amy 账号都给这条微博点了赞，显示点赞数
- 编辑用户个人信息并能通过点击用户名字显示

搜索

amy

修改信息

退出登录

用户详细信息

性别

受教育程度

职业

住址

手机号码

电子邮箱

个人简介

保存

搜索

amy 修改信息

性别: 女

教育程度: 本科

工作: 学生

地址: 学校

个人简介: 12412451

电话号码: 4315435

邮箱: 124124



点击这个名字也可以跳转到详情信息界面。

➤ 关注功能:

首页 发布微博 开始搜索吧 搜索

amy 修改信息

微博广场 体育 娱乐 学习 新闻 日常生活 社交 关注 关注用户 amy的微博

mike

关注 运动打卡~

2021-04-24 13:52:05

转发 评论 点赞 (2)

评论列表

对于 amy 用户而言, mike 是可以关注的对象, 而 amy 不能关注自己:

首页 发布微博 开始搜索吧 搜索

amy 修改信息 退出登录

微博广场 体育 娱乐 学习 新闻 日常生活 社交 关注 关注用户 amy的微博

amy

今天很开心!

2021-04-24 23:33:31

转发 评论 点赞 (2)

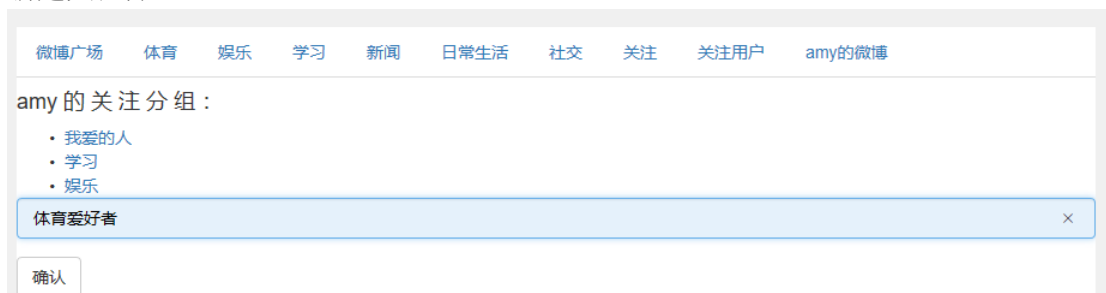
点击关注后, 可以选择将其放入哪个分组:



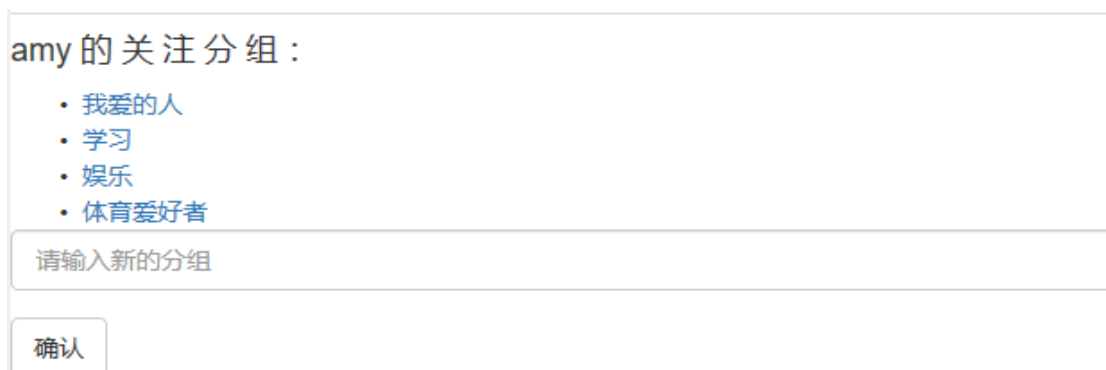
选好分组后点击返回，关注按钮将变成已关注：



#### ➤ 新建关注分组



点击确认后：



点击每个分组的超链接可以显示分组里关注的用户：



点击 mike 的超链接就能进入他的个人详细信息界面。

## ➤ 搜索文章功能



点击搜索后:



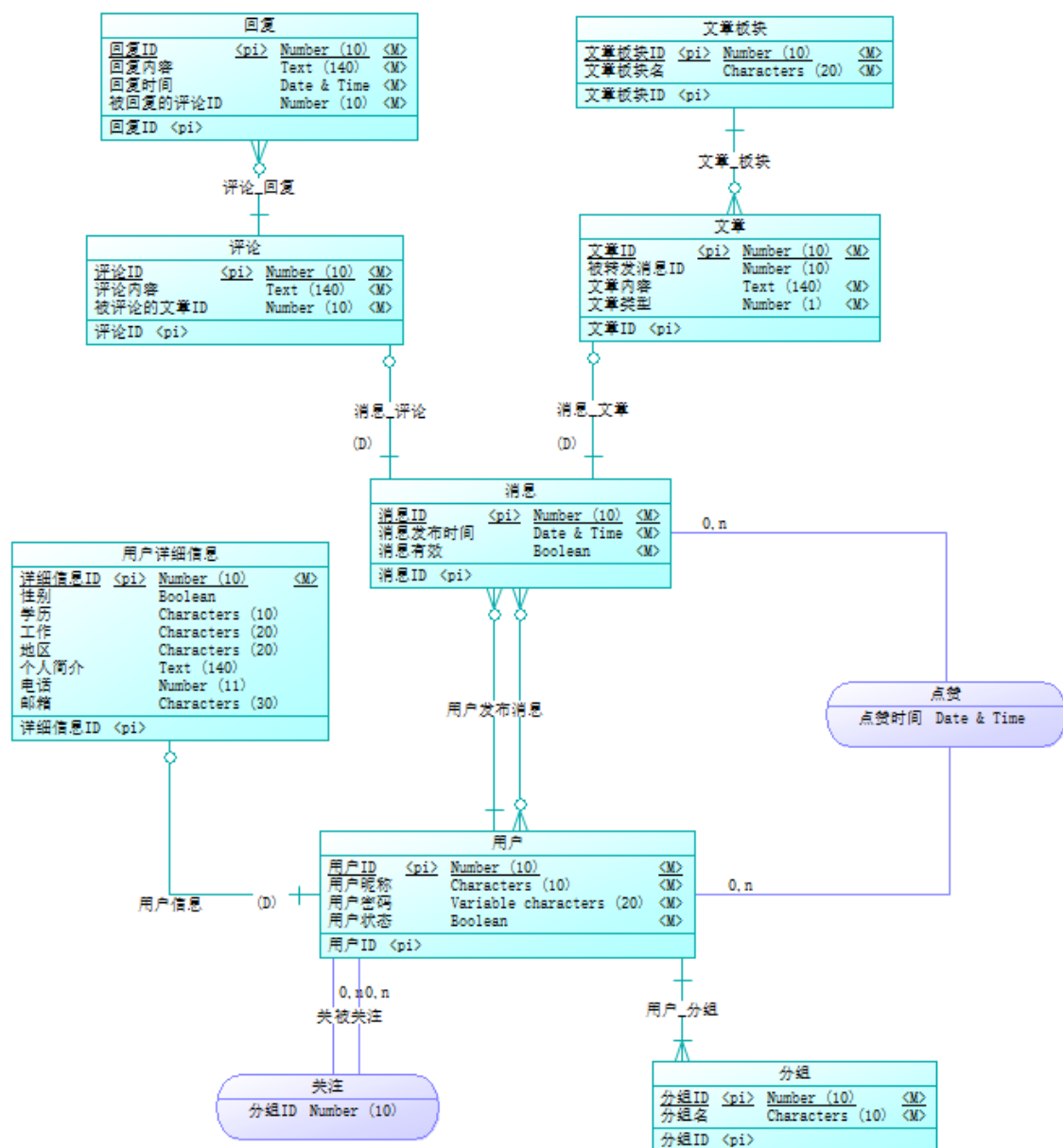
此处测试只限于一个用户的微博，实则可以搜索整个微博系统中的所有微博，且按照发布时间倒序排列，最近发布的排在最上。

## 2.2 数据库设计

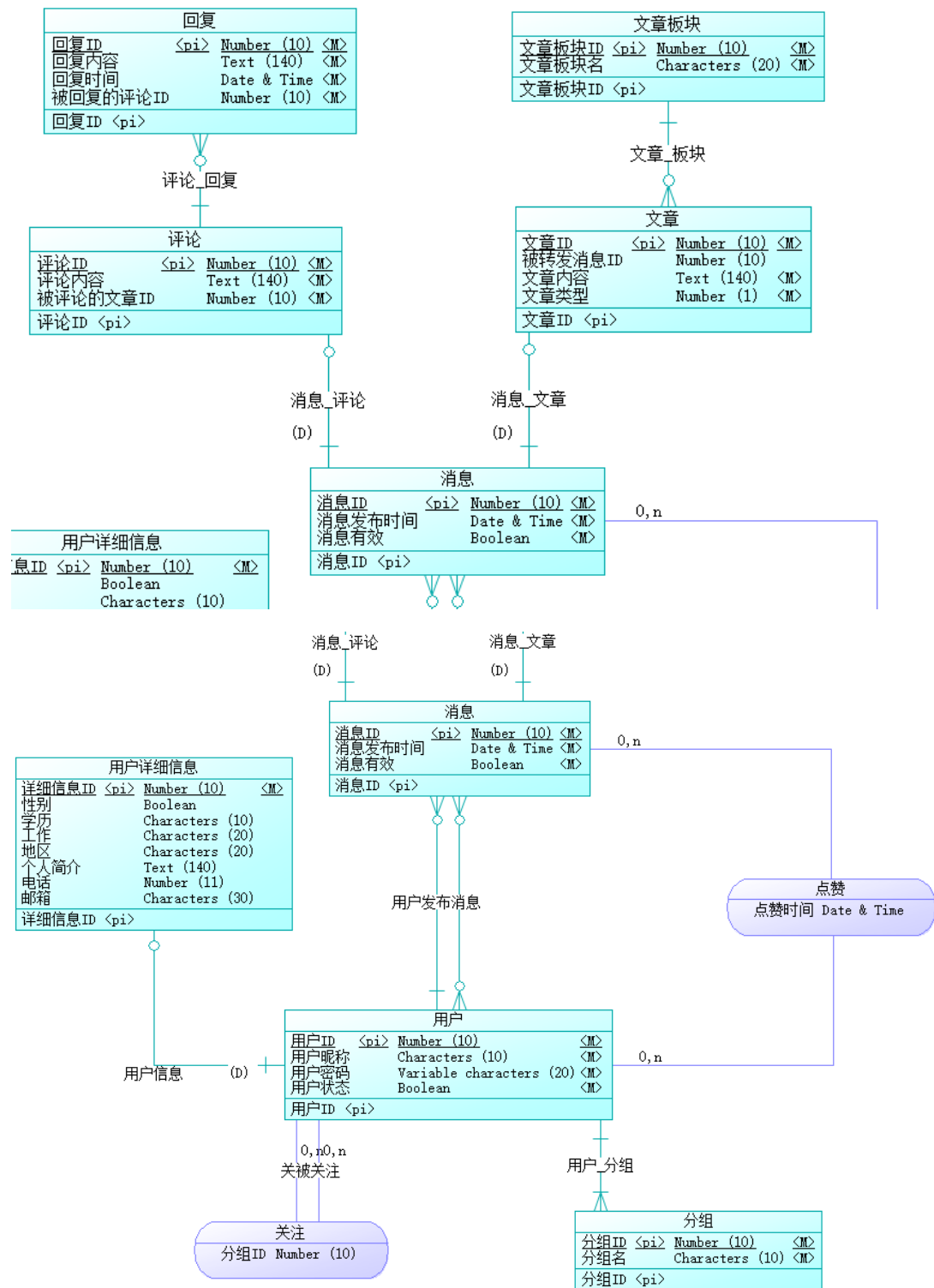
。

### 2.1.1 ER 图

要求：截图务必清晰，如果图太大可截图一个总图，然后截几个部分图展示。如果看不清截图会影响成绩。

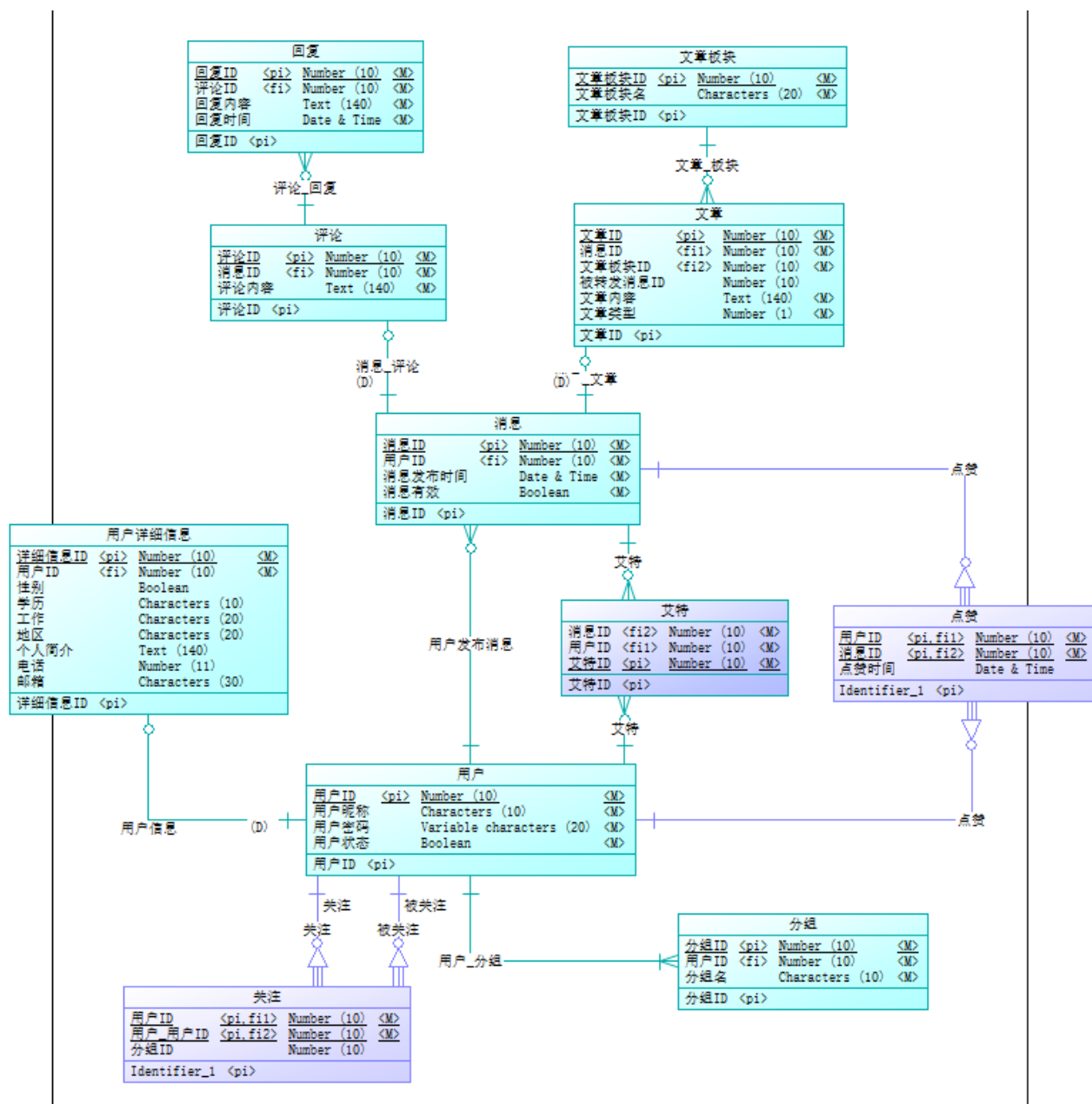


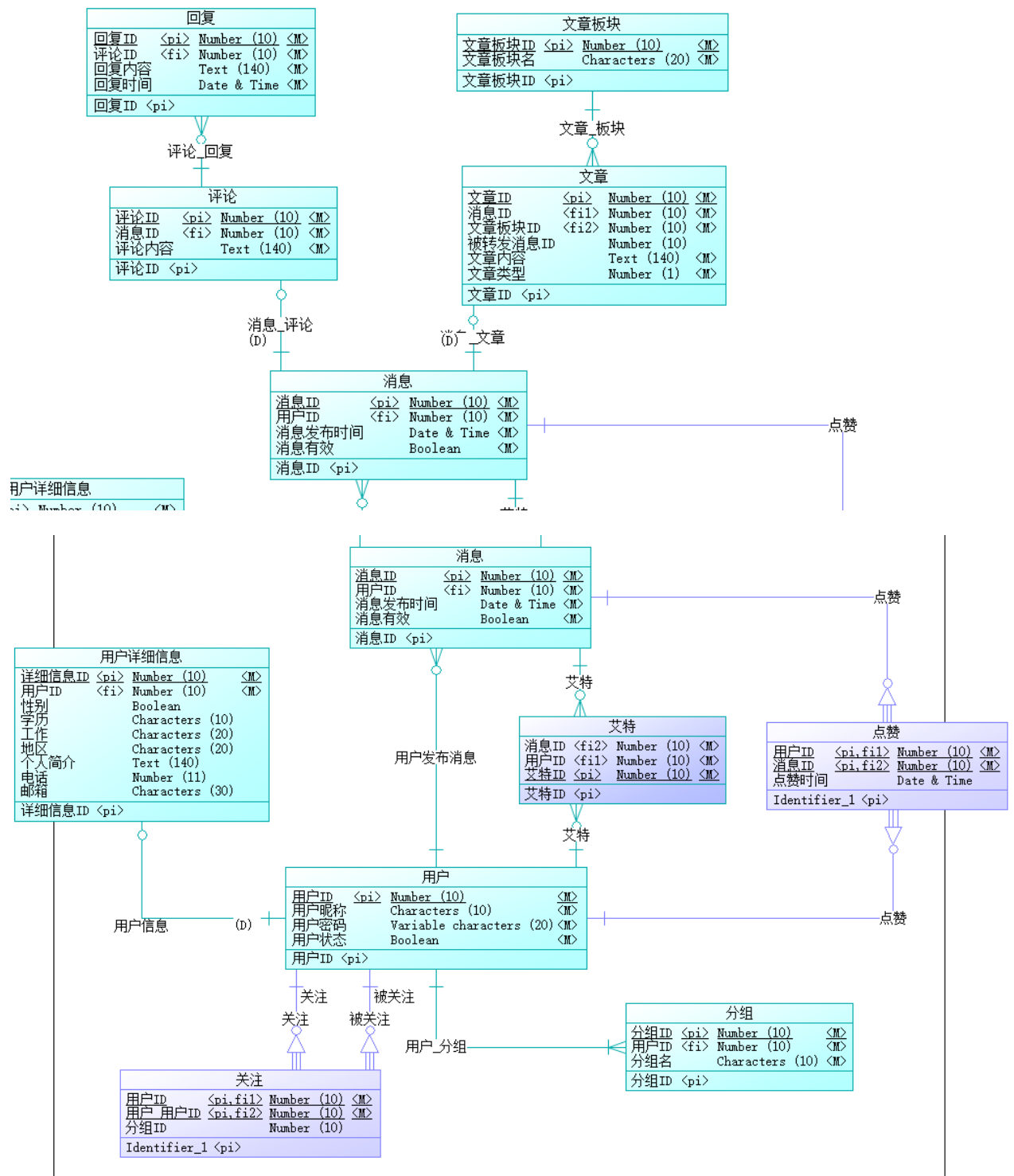




## 2.1.2 LDM 图

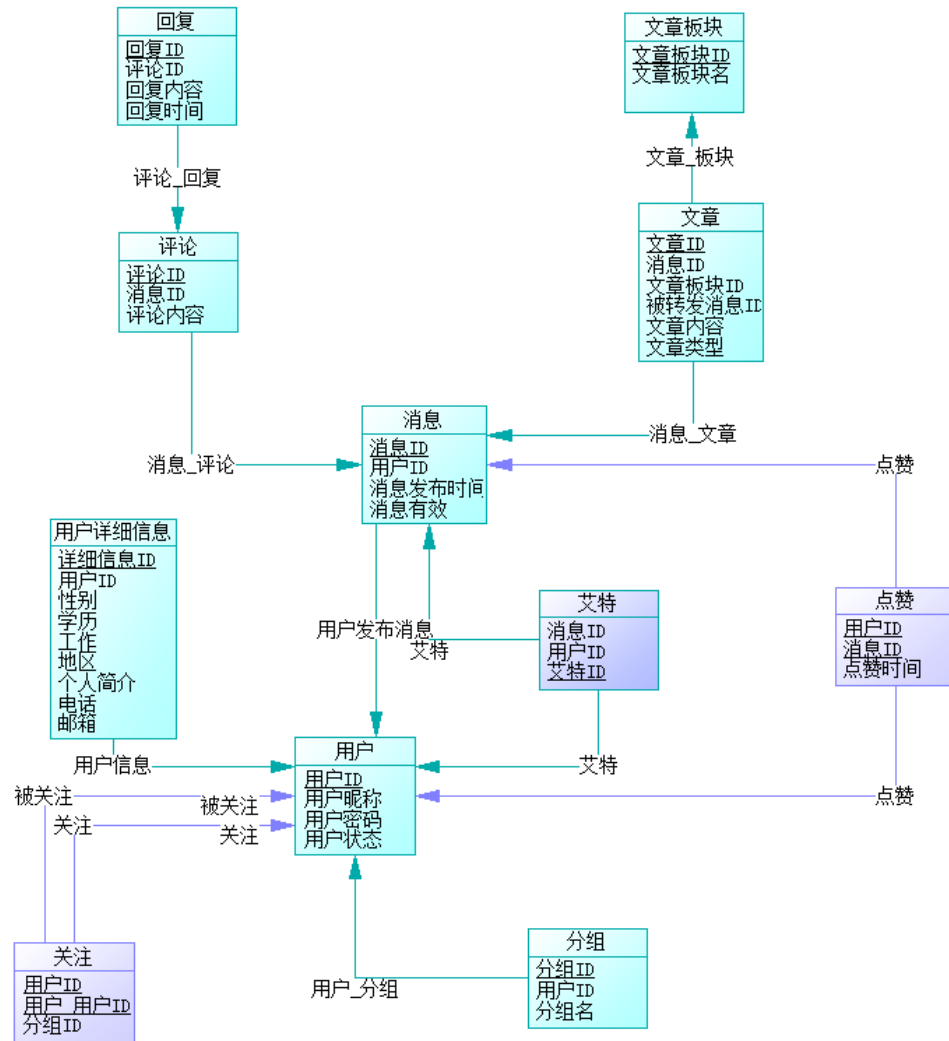
要求：截图务必清晰，如果图太大可截图一个总图，然后截几个部分图展示。如果看不清截图会影响成绩。





## 2.1.3 PDM 图

要求：截图务必清晰，如果图太大可截图一个总图，然后截几个部分图展示。如果看不清截图会影响成绩。



### 消息内容汇总

用户ID  
 消息发布时间  
 消息有效  
 文章板块ID  
 文章内容  
 文章类型  
 被转发消息ID  
 用户昵称  
 评论内容  
 praise\_num  
 comment\_num

☐ message1  
☐ article  
☐ comments  
☐ userinfo

☐ (select praise.message\_id as message\_id, count(p  
☐ (select comments.message\_id as message\_id, count



## 2.1.4 数据库表结构


### 1、 表结构

选取几个比较有代表性的表结构截图，体现主键约束、外键约束、空值约束。

```
/*=====*/
/* Table: userinfo */
/*=====*/
create table userinfo
(
    user_ID          int not null auto_increment,
    user_name        char(10) not null unique,
    user_psw         varchar(20) not null,
    user_state       bool not null,
    primary key (user_ID)
);
```

Result Grid



Filter Rows:

Edit: 

	user_ID	user_name	user_psw	user_state
▶	1	amy	1234	1
	2	mike	1234	1
	3	jack	1234	1
✱	NULL	NULL	NULL	NULL

主键为 user\_ID，每个值都非空。

```
/*=====*/
/* Table: praise */
/*=====*/
create table praise
(
    user_ID          int not null ,
    message_ID       int not null,
    praise_time       datetime,
    primary key (user_ID, message_ID)
);
```

Result Grid			
		Filter Rows:	
Edit:			
	user_ID	message_ID	praise_time
▶	1	2	2021-04-24 14:53:08
	1	3	2021-04-24 14:53:50
	1	7	2021-04-24 23:39:30
	2	3	2021-04-24 23:03:22
	3	1	2021-04-24 13:54:31
	3	2	2021-04-24 13:54:23
	3	3	2021-04-24 13:54:23
	3	7	2021-04-24 23:39:24
★	NULL	NULL	NULL

主键为 message\_ID 和 user\_ID，点赞时间用处不算大，可以为空，其他都为非空。

```

/*=====*/
/* Table: article */
/*=====*/

create table article
(
    article_ID          int not null auto_increment,
    message_ID          int not null,
    section_ID          int not null,
    t_article_ID        int,
    article              text not null,
    article_type         int not null,
    primary key (article_ID)
);

```

Result Grid						
		Filter Rows:		Edit:		Export/Import:
	article_ID	message_ID	section_ID	t_article_ID	article	article_type
▶	1	1	6	0	早起学习打卡~	0
	2	2	5	0	运动打卡~	0
	3	3	2	0	哈哈交朋友呀	0
	4	7	1	0	今天很开心!	0
★	NULL	NULL	NULL	NULL	NULL	NULL

主键为 article\_ID，外键为 message\_ID，除了被转发的文章 ID 可以为空，其他属性不允许空值。

## 2、索引

*讲解你建的非主索引，说明为什么要建这个索引。*

➤ 非主属性索引：userinfo 表中的 user\_name

- ✧ 之所以建这个索引，是因为在用户界面显示的总是用户名，而其他表中记录的用户信息都是 user\_ID，也就是外键，因此建立这个索引可以借助 session 获取的用户名来获取该用户的其他信息，如关注、发布的微博、评论等。

### 3、 视图

讲解你建的视图，说明为什么要建这个视图。

- 建的视图主要作用是统计点赞数。

点赞表的结构如下：

Result Grid			
Filter Rows:			
	user_ID	message_ID	praise_time
▶	1	1	2021-04-25 14:32:20
	2	1	2021-04-25 14:37:55
	3	1	2021-04-25 14:32:49
*	NULL	NULL	NULL

记录的是点赞的用户、被点赞消息的 ID 以及点赞的时间，而在每条微博之中，需要显示的是点赞的数量，因此需要建一个视图来表现点赞的数量。

### 4、 触发器（选做）

讲解你为什么建触发器和如何实现触发器。

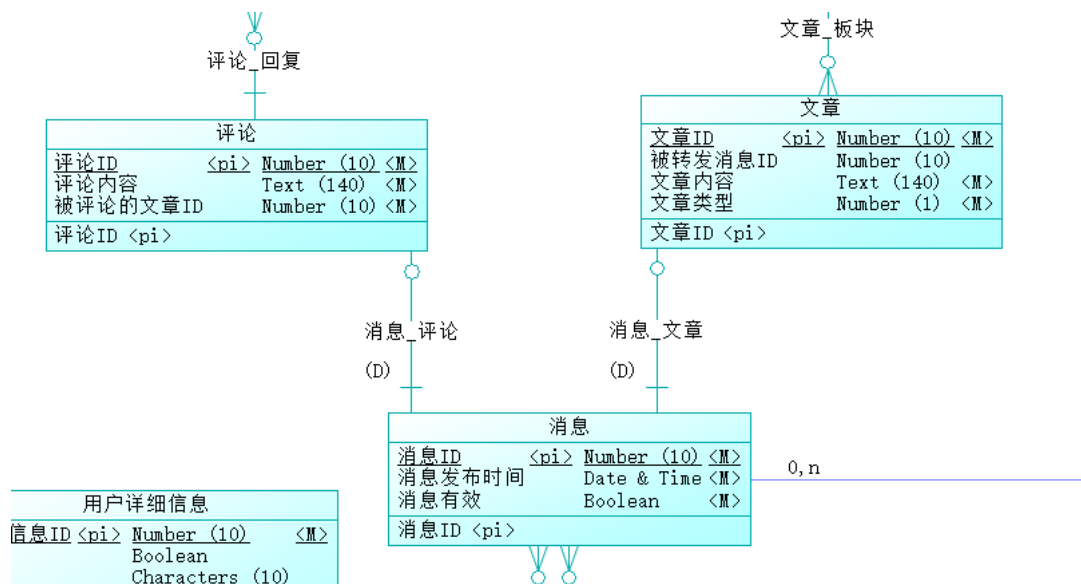
### 5、 事务（选做）

讲解你为什么建事务和如何实现事务。

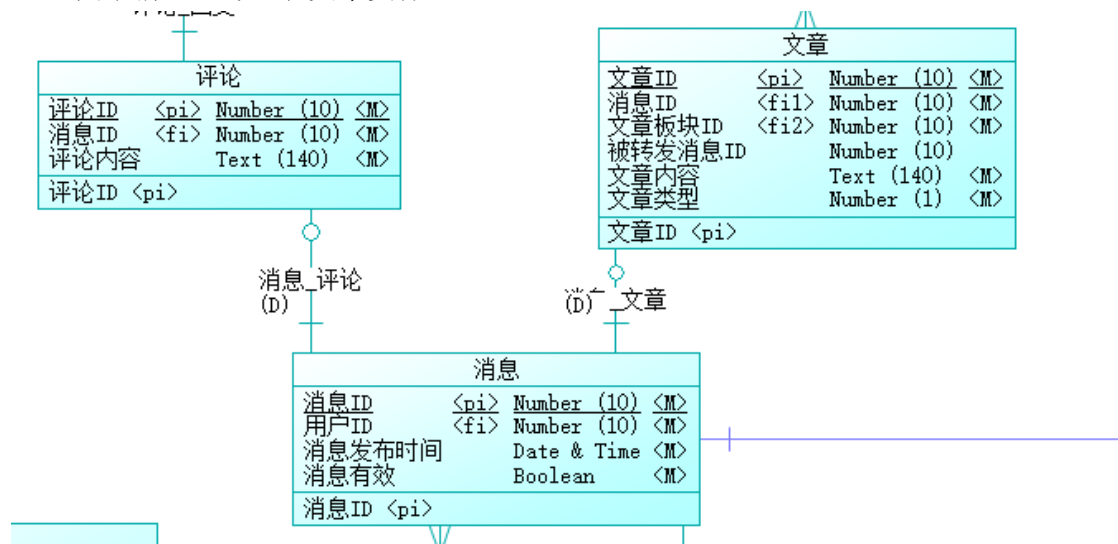
## 2.1.5 分析

选择较为有代表性的实体和联系进行分析，讲解如何从 ER 图到 LDM 图、再到 PDM 图，最后变成数据库表结构的转换过程（参考实验二 PPT 里“补充知识 三、ER 图→LDM 图→PDM 图”中老师讲的例子）。

ER 图中消息、评论和文章实体

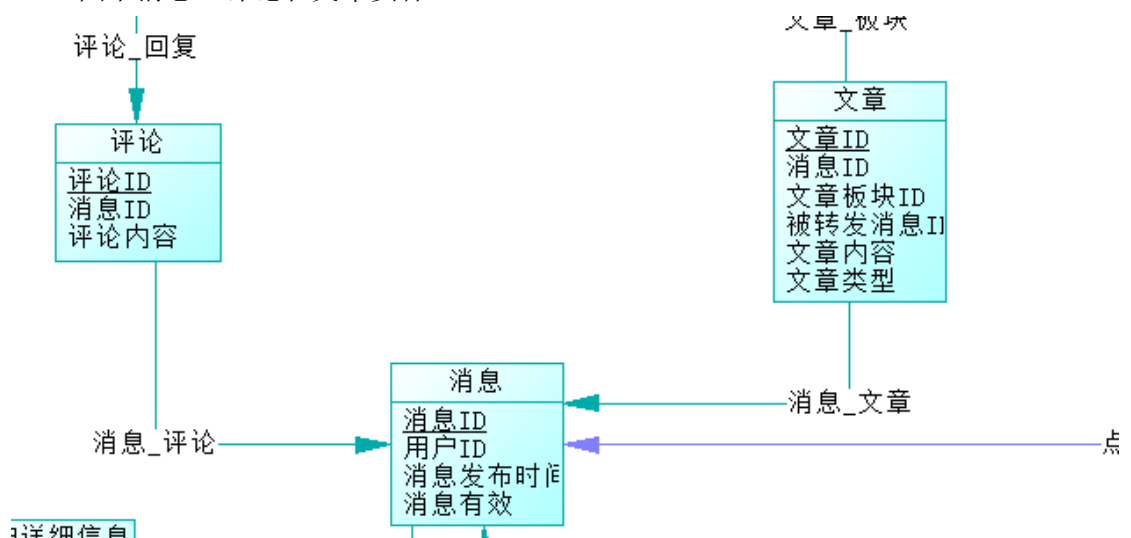


LMD 图中消息、评论和文章实体



ER 图中所设置的外键出现在了消息、评论和文章实体中。

PDM 图中消息、评论和文章实体



详细信息

数据库中的表:



消息实体转化的数据库表：

	message_ID	user_ID	message_time	valid
▶	1	1	2021-04-25 14:32:02	1
	2	1	2021-04-25 14:32:31	1
	3	3	2021-04-25 14:32:59	1
	4	2	2021-04-25 14:34:27	1
	5	1	2021-04-25 15:24:02	1
*	NULL	NULL	NULL	NULL

评论实体转化的数据库表：

	comment_ID	message_ID	comment_text	c_message_ID
▶	1	2	你好，我叫jack	1
	2	3	你好，我叫mike	1
	3	4	你好，我叫jack	1
*	NULL	NULL	NULL	NULL

新增了一个 c\_message\_ID,表示被评论的 message 的 ID，其他属性都和 ER 表、LDM 表和 PDM 表中的相同。

文章实体转化的数据库表：

	artide_ID	message_ID	section_ID	t_artide_ID	artide	artide_type
▶	1	1	2	0	大家好，我叫amy	0
	2	5	6	0	学习数据库ing	0
*	NULL	NULL	NULL	NULL	NULL	NULL

从上述图中，我们可以看到，ER 图中的属性，在数据库表中都有所体现；LDM 图、PDM 图中写出的外键，也能在数据库表中自动生成属性，方便将表进行连接。

### 3 收获和反思

*请填写本次实验的收获，记录实验过程中出现的值得反思的问题及你的思考。*

**收获：**“拎包入住”实验室后的一周，终于完成了这个微博系统所要求的功能，还有一些实体中的功能没有精力实现了。一共实现了 9 个实体功能，分别是：用户注册信息、用户基本信息、用户发送的文章、用户发送的评论、用户的点赞、用户的关注、文章所在的模块、文章和评论统称为消息、用户所建立的关注分组，能够基本完成 ppt 上的系统要求。

这是第一次接触 html 和网页的编程，也是第一次接触 flask 框架的 web 编程。经过这次实验，不仅熟悉了 web 编程，还学会了与后台数据库的连接，在一次次美化页面、优化功能后，都有十足的满足感。经过从 0 开始的摸爬滚打，到一个初具雏形的小型微博系统，我的编程能力和处理问题的能力又有了提高。

**值得反思的问题及我的思考：**此次实验的过程中，没有注意及时写下注释，以后要不断完善；代码重复率较高，耦合度不够，以后要学会改善；此次编写网页遇到最多次的错误是 500，主要原因是 cursor 连接数据库操作的出错，经过好多次的错误后才渐渐摸清正确的语法和思路；有一些自己理想的按键反应、页面刷新方案没法实现，因为我现有的对 html 的了解还不够深入，js 和 css 语言还不太熟悉，因此很多跳转和刷新只能使用最朴素的做法实现。